IN THE DRAWINGS

Figure 7 was objected to because element 709 is not shown. The objection is believed addressed in the "In The Specification" section of the present response.

REMARKS

Claims 1-30 are pending. Claims 1-16 and 20-30 were rejected. Claims 17-19 were objected to but allowable if amended to include base and intervening claim limitations. The allowability of claims 17-19 is gratefully acknowledged. Claims 6, 8, 26, and 28 were objected to because of informalities. Claims 4 and 24 were amended to address antecedent basis informalities in claims 6, 8, 26, and 28.

Claims 1-16 and 20-30 were rejected under 35 U.S.C. 102(e) as being anticipated by Frankel (USPAP 20030093254). Frankel describes a distributed simulation system "which includes at least a first node and a second node. The first node is configured to simulate a first portion of a system under test using a first simulation mechanism. The second node is configured to simulate a second portion of the system under test using a second simulation mechanism different from the first simulation mechanism. The first node and the second node are configured to communicate during a simulation using a predefined grammar." [0007] "The models are frequently described in a hardware description language (HDL) such as Verilog, VHDL, etc. The HDL model may be simulated in a simulator designed for the HDL, and may also be synthesized, in some cases, to produce a netlist and ultimately a mask set for fabricating an integrated circuit." [0004]

By contrast, claims 1, 20, and 30 recite receiving a "high-level language program" and identifying a portion of the "high-level language program." An HDL model is not a high-level language program.

As stated in the specification of the present application, "A system on a programmable chip typically includes logic implemented using a Hardware Description Language (HDL). However, using HDL may not be a very efficient or effective way of optimizing an application optimized programmable chip. HDL often requires knowledge about the underlying hardware, and relatively few people have familiarity with HDL. Other languages for implementing logic on a device have been developed based on high-level programming languages. Any language that can be used to describe software functions and/or objects without extensive knowledge of the underlying hardware used to implement the software is referred to herein as a high-level language. Examples of high-level languages include C, C++, Java, and Pascal conventionally used by software engineers. Other variations include Matlab and VisualBasic. High-level

languages are typically general purpose and interact with hardware using operating system associated application program interfaces (APIs)." (page 6, lines 20-33) Frankel does not teach receiving any high-level language program or identifying any portion of a high-level language program.

Claims 1, 20, and 30 also recite "the high-level language program configured to run on a conventional central processing unit." HDL is not configured to run on a conventional central processing unit. A conventional central processing unit does not understand HDL or typical compile results generated using HDL. HDL is used to implement designs on programmable chips, but HDL can not be configured to run on a conventional central processing unit.

CONCLUSION

In light of the above remarks, the rejections to the independent claims are believed overcome for at least the reasons noted above. Applicants believe that all pending claims are allowable in their present form. Please feel free to contact the undersigned at the number provided below if there are any questions, concerns, or remaining issues.

Respectfully submitted,
BEYER WEAVER & THOMAS, LLP

Godfrey K. Kwan
Reg. No. 46,850

P.O. Box 70250
Oakland, CA 94612-0250
(510) 663-1100